

AD-A173 644

COMPUTER CENTER (VAXCLUSTER LIBRARIES/UTILITY
(PROGRAMS)(U) DAVID W TAYLOR NAVAL SHIP RESEARCH AND
DEVELOPMENT CENTER BET D V SOMMER MAY 86

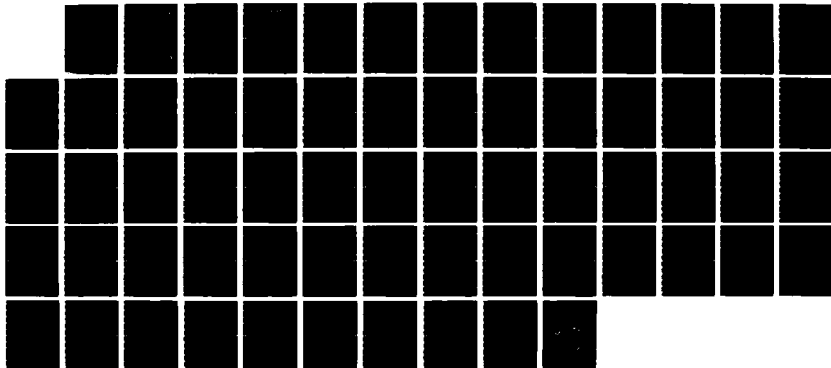
1/1

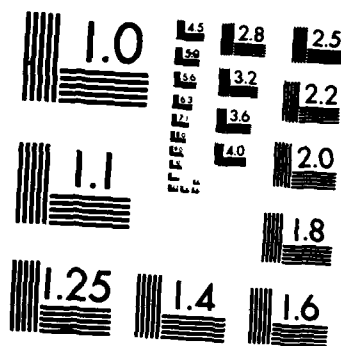
UNCLASSIFIED

DTNSRDC/CNLD-TM-18-86-15

F/G 9/2

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

TM-18-86-15

Computer Center VAXcluster Libraries/UTILITY (Programs)

DAVID W. TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Maryland 20084



AD-A173 644

COMPUTER CENTER
VAXCLUSTER LIBRARIES/UTILITY
(PROGRAMS)

by

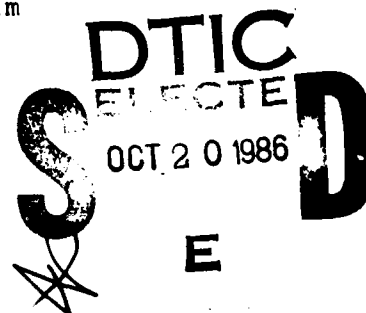
DAVID V. SOMMER

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

DTIC FILE COPY

Computation, Mathematics and Logistics Department
Technical Memorandum

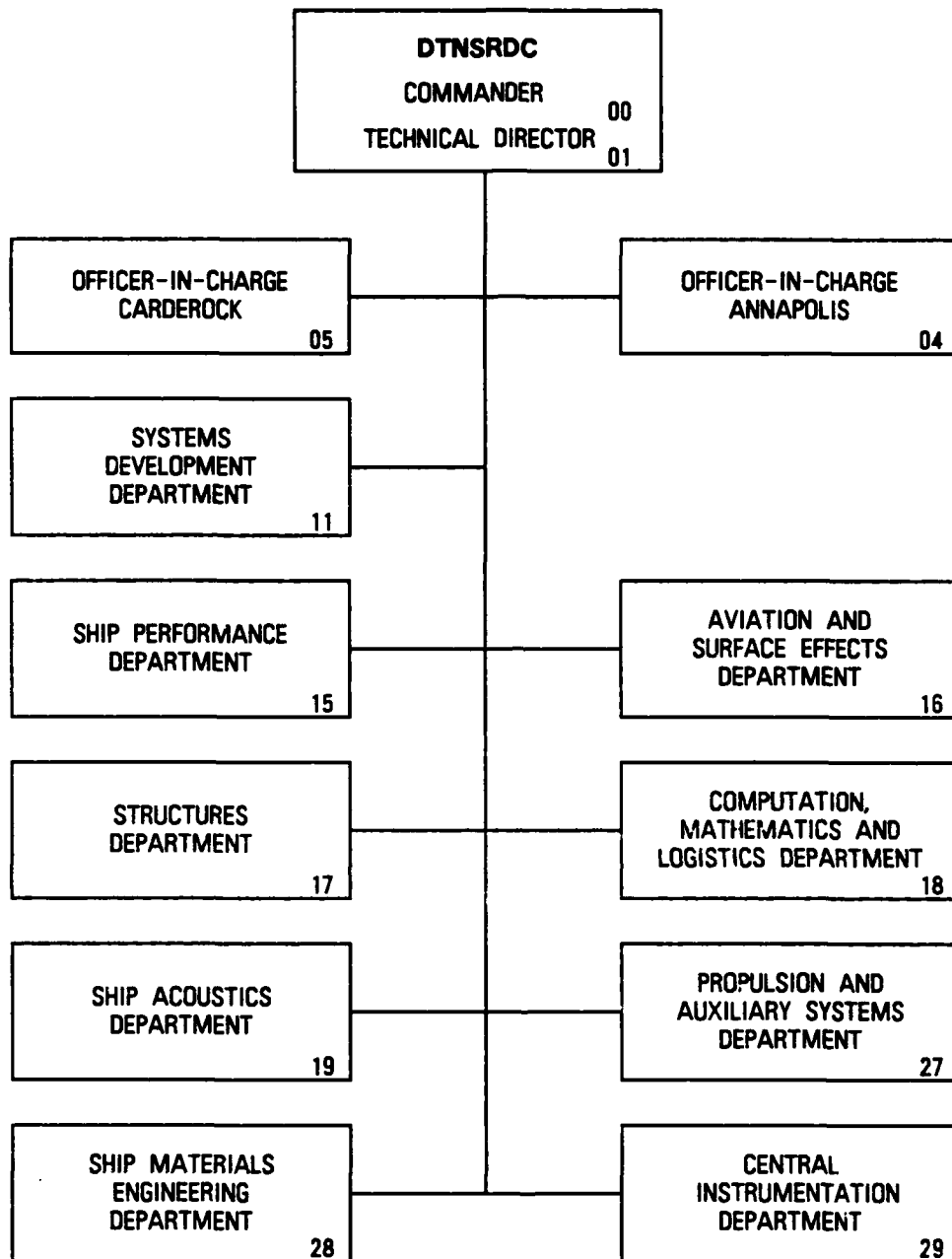
May 1986



TM-18-86-15

406847 PTM

MAJOR DTNSRDC ORGANIZATIONAL COMPONENTS



REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER TM-18-86-15	2. GOVT ACCESSION NO. AD-A173644	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Computer Center DEC VAXcluster Libraries: Utility (Programs)		5. TYPE OF REPORT & PERIOD COVERED
7. AUTHOR(s) David V. Sommer		6. PERFORMING ORG. REPORT NUMBER Final
9. PERFORMING ORGANIZATION NAME AND ADDRESS DTNSRDC, User Services, Code 1892 Bethesda, Maryland 20084-5000		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Computation, Mathematics & Logistics Dept. Computer Facilities Division (189)		12. REPORT DATE May 1986
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES 63
		15. SECURITY CLASS. (of this report) Unclassified
16. DISTRIBUTION STATEMENT (of this Report) Approved for Public Release; Distribution Unlimited		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Computers Software documentation Control statements Utilities DEC VAXcluster Functional categories		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) > The Computer Center DEC VAXcluster Libraries: Utility (Programs), VLIB/U, is a reference manual which describes the public-access programs developed at DTNSRDC for the DEC VAXcluster. These edited on-line helps include listers, file converters, program analyzers, etc. In addition, VLIB/U lists them by functional category and alphabetically with a descriptive title for each.		

David W. Taylor
Naval Ship Research and Development Center
Bethesda, Maryland 20084-5000

*
*
* Computer Center
* DEC VAXcluster
* Libraries: UTILITY
* (Programs)
*
*



by
David V. Sommer

User Services Branch
Code 1892

	Carderock	Annapolis
Phone	(202) 227-1907	(301) 267-3343
Autovon	287-1907	281-3343

Accession For		
NTIS GRA&I	<input checked="" type="checkbox"/>	
DTIC TAB	<input type="checkbox"/>	
Unannounced	<input type="checkbox"/>	
Justification		
By _____		
Distribution/ _____		
Availability Codes		
1	2	3
4	5	6
7	8	9
10	11	12
Special		
A-1		

For recorded message on computer status (202) 227-3043

Questions and requests for more detailed information
should be directed to Code 1892, Bldg. 17, Rm. 100.

Computation, Mathematics and Logistics Department
Technical Memorandum

May 1986

TM-18-86-15

Table of Contents

1	Introduction	
	What's In This Manual	1-1
	Contents	1-2
	Functional Categories	1-4
	List of Routines by Category	1-7
2	Routine documentation	
	How to Get On_line Help	2-1
	<individual HELP modules arranged alphabetically>	2-2 *

*** How This Was Prepared ***

This is a printed document of the on-line help modules available. There has been no attempt to "neaten" them up -- the spacing is as it was designed to be displayed by the VMS HELP program. A procedure and a program were written to extract, arrange and format them.

* - As new routines are developed, the HELP modules may be printed and inserted into this document.

***** Introduction *****

The Computer Center makes available on VAXcluster, in addition to the VMS operating system, a wide variety of both scientific and utility programs, subprograms and procedures. The routines are maintained in libraries or as separate files in the VSYS: directory.

The VLIB-Series consists of the following, which are the helps for the various VAXcluster "libraries" maintained by the Computer Center:

VLIB/D - Computer Center VAXcluster Libraries / DTNSRDC (Commands and General Information)	TM-18-86-12
VLIB/N - Computer Center VAXcluster Libraries / NSRDC (Subprograms)	TM-18-86-13
VLIB/P - Computer Center VAXcluster Libraries / PROCFIL (Procedures)	TM-18-86-14
VLIB/U - Computer Center VAXcluster Libraries / UTILITY (Programs)	TM-18-86-15

*** What's In This Manual ***

A list of the routines with a brief description of each is followed by the list of functional categories used to classify each routine. Next is a list of the routines under the various categories. Chapter 2 contains the currently available HELP modules in alphabetical order.

**** Contents ****

The following utility programs were written at DTNSRDC. For help, type "HELP @UTILITY routine".

AUX	Turn an auxiliary printer (one attached to an interactive terminal) on or off.
BANNER	Write 10-line-high banners, up to three banners per page, entered one banner at a time.
BANNER3	Write 10-line-high banners, up to three banners per page, entered one page (3 banners) at a time.
BANNER6	Write 6-line-high banners, up to six banners per page, entered one page (6 banners) at a time.
By_Category	List of modules by the functional category to which each belongs.
By_Date	List of modules in reverse order by the date of the last modification to the module or its help.
CALLTREE	Produce a flow chart of subroutine and function calls for a FORTRAN program.
CMP2FOR	Prepare compiler listing for printing on CDC.
COPYSF	Copy shifted file.
CPYEXT	Copy a file and extract fields. Useful for removing trailing id field in a source program, re-arranging data fields, etc.
DETAB	Remove tabs from a file or convert tab-format Fortran source lines to fixed-format.
HEAD	Display the first 22 lines of a file.
LIST1	Produce a centered list of a document or form (up to 90 characters per line and having carriage control in column 1). Optionally, the record-number-in-file, page number, line-number-on-page, and line lengths may be printed. Lines that are too long may also be flagged.
LISTM	List a file in multiple columns.
LISTN	Numbered list of a file. Optionally, the line lengths may be listed and lines longer than a specified length may be flagged.
LST2FOR	Copy file changing record attribute from "Carriage return carriage control" to "Fortran carriage control".
LPL	List page lengths of document or manual.
MAKSUB	Generate a skeleton Fortran program or subprogram.

TAIL Display the last 22 lines of a file.

UNBLANK Removes trailing blanks from a file. This is useful for
 shortening files moved to VMS from another computer.

UNDECK Break up a CDC UPDATE source file into separate files.

VLASER Combine files for Xerox 8700.

*** Functional Categories ***

The following functional categories are used at DTNSRDC. Those preceded by an asterisk (*) are local DTNSRDC categories. All others are from VIM (the CDC users group).

- A0 Arithmetic routines
 - A1 Real numbers
 - A2 Complex numbers
 - A3 Decimal
 - A4 I/O routines
- B0 Elementary functions
 - B1 Trigonometric
 - B2 Hyperbolic
 - B3 Exponential and logarithmic
 - B4 Roots and powers
- C0 Polynomials and special functions
 - C1 Evaluation of polynomials
 - C2 Roots of polynomials
 - C3 Evaluation of special functions (non-statistical)
 - C4 Simultaneous non-linear algebraic equations
 - C5 Simultaneous transcendental equations
 - * C6 Roots of functions
- D0 Operations on functions and solutions of differential equations
 - D1 Numerical integration
 - D2 Numerical solutions of ordinary differential equations
 - D3 Numerical solutions of partial differential equations
 - D4 Numerical differentiation
- E0 Interpolation and approximations
 - E1 Table look-up and interpolation
 - E2 Curve fitting
 - E3 Smoothing
 - E4 Minimizing or maximizing a function
- F0 Operations on matrices, vectors & simultaneous linear equations
 - F1 Vector and matrix operations
 - F2 Eigenvalues and eigenvectors
 - F3 Determinants
 - F4 Simultaneous linear equations
- G0 Statistical analysis and probability
 - G1 Data reduction (common statistical parameters)
 - G2 Correlation and regression analysis
 - G3 Sequential analysis
 - G4 Analysis of variance
 - G5 Time series
 - G6 Special functions (includes random numbers and pdf's)
 - * G7 Multivariate analysis and scale statistics
 - * G8 Non-parametric methods and statistical tests
 - * G9 Statistical inference

H0 Operations research techniques, simulation & management science

- H1 Linear programming
- H2 Non-linear programming
- H3 Transportation and network codes
- H4 Simulation modeling
- H5 Simulation models
- H6 Critical path programs
- H8 Auxiliary programs
- H9 Combined

I0 Input

- I1 Binary
- I2 Octal
- I3 Decimal
- I4 BCD (Hollerith)
- I9 Composite

J0 Output

- J1 Binary
- J2 Octal
- J3 Decimal
- J4 BCD (Hollerith)
- J5 Plotting
- J7 Analog
- J9 Composite

K0 Internal information transfer

- K1 External-to-external
- K2 Internal-to-internal (relocation)
- K3 Disk
- K4 Tape
- K5 Direct data devices

L0 Executive routines

- L1 Assembly
- L2 Compiling
- L3 Monitoring
- L4 Preprocessing
- L5 Disassembly and derelativizing
- L6 Relativizing
- L7 Computer language translators

M0 Data handling

- M1 Sorting
- M2 Conversion and/or scaling
- M3 Merging
- M4 Character manipulation
- M5 Searching, seeking, locating
- M6 Report generators
- M9 Composite

N0 Debugging

- N1 Tracing and trapping
- N2 Dumping
- N3 Memory verification and searching
- N4 Breakpoint printing

- 00 Simulation of computers and data processors (interpreters)
 - 01 Off-line equipment (listers, reproducers, etc.)
 - 03 Computers
 - 04 Pseudo-computers
 - 05 Software simulation of peripherals
 - 09 Composite
- P0 Diagnostics (hardware malfunction)
- Q0 Service or housekeeping, programming aids
 - Q1 Clear/reset
 - Q2 Checksum accumulation and correction
 - Q3 File manipulation
 - Q4 Internal housekeeping, save, restore, etc.
 - Q5 Report generator subroutines
 - Q6 Program documentation: flow charts, document standardization
 - Q7 Program library utilities
- R0 Logic and symbolic
 - R1 Formal logic
 - R2 Symbol manipulation
 - R3 List and string processing
 - R4 Text editing
- S0 Information retrieval
- T0 Applications and application-oriented programs
 - T1 Physics (including nuclear)
 - T2 Chemistry
 - T3 Other physical sciences (geology, astronomy, etc.)
 - T4 Engineering
 - T5 Business data processing
 - T6 Manufacturing (non-data) processing and process control
 - T7 Mathematics and applied mathematics
 - T8 Social and behavioral sciences and psychology
 - T9 Biological sciences
 - T10 Regional sciences (geography, urban planning)
 - T11 Computer assisted instruction
- U0 Linguistics and languages
- V0 General purpose utility subroutines
 - V1 Random number generators
 - V2 Combinatorial generators: permutations, combinations & subsets
 - * V3 standard and special problems
- X0 Data reduction
 - X1 Re-formatting, decommutation, error diagnosis
 - X2 Editing
 - X3 Calibration
 - X4 Evaluation
 - X5 Analysis (time-series analysis)
 - X6 Simulation (generate test data for data reduction system)
- Y0 Installation modification
 - Y1 Installation modification library
 - Y2 NEWPL tape of installation modifications
- Z0 All others

**** By Functional Category ****
(14-MAY-86 @ 10:17:50)

The modules in this library are listed below by functional category.

(E - executable program; F - function subprogram; P - procedure;
S - subroutine subprogram; Z - miscellaneous)

J4 BCD (Hollerith)

E-BANNER E-BANNER3 E-BANNER6

K1 External-to-external

E-COPYSF E-CPYEXT

M0 Data handling

E-COPYSF E-CPYEXT E-UNBLANK E-UNDECK

M2 Conversion and/or scaling

E-LST2FOR

O1 Off-line equipment (listers, reproducers, etc.)

E-CMP2FOR E-HEAD E-LIST1 E-LISTM E-LISTN
E-TAIL E-VLASER

Q0 Service or housekeeping, programming aids

E-CALLTREE E-LPL E-MAKSUB E-VLASER

***** Individual Documents *****

This chapter contains the HELP modules for all routines and general information in "library" UTILITY.

For the most recent on-line HELPs, type

HELP @UTILITY <routine>

To see the current contents, type

HELP @UTILITY Contents

To see the most recently changed routines of HELPs, type

HELP @UTILITY By_Date

To see the current functional category list of the modules, type

HELP @UTILITY By_Category

**** BANNER ****

Write 10-line-high banners of up to 10 characters each (up to 3 per page).

Format:

RUN VSYS:BANNER

text /OUTPUT=file_spec /[NO]NEWPAGE <-- one or more of these,
end with empty line

Up to three banners can be printed on one page. This is designed for printing on wide paper.

See BANNER3 for 10-line-high banners entered one page (3 banners) at a time, and BANNER6 for 6-line-high banners for narrow paper.

*** Data ***

RUN VSYS:BANNER

text /OUTPUT=file_spec /[NO]NEWPAGE

You will be prompted for the message text. Each data entry has the form:

text /OUTPUT=file_spec /[NO]NEWPAGE

where text is up to 10 characters. The following are interpreted specially:

NODE - DT1 or DT2, as appropriate

DATE - the date in VAX/VMS format (dd-mmm-yyyy)

TIME - the time (hh:mm:ss)

Default: <no text> (a blank banner)

*** Qualifiers ***

/NEWPAGE

/NEWPAGE

/NONEWPAGE

/NEWPAGE causes the banner to be written on a new page.

/NONEWPAGE causes the banner to be written on the current page.

Default: /NEWPAGE for the first message

/NONEWPAGE for succeeding messages

/OUTPUT

/OUTPUT=file_spec

Specifies the file into which the banner is to be written.

Default: /OUTPUT=SYSS\$OUTPUT for the first message.

The previous file_spec is used for succeeding messages (that is, messages are written to the same file until changed).

*** Examples ***

RUN VSYS:BANNER

text /OUTPUT=file_spec /[NO]NEWPAGE

- 1) Write the banners "MYOWN" and "LIBRARY", list the library and send to the printer.

RUN VSYS:BANNER

MYOWN /OUTPUT=A.LIS <-- written on a new page

LIBRARY <-- written on the same page

<cr> <-- no more banners (empty line)

LIBRARY/LIST/FULL/OUTPUT=A.LIS myown_library <-- make library list

PRINT/NOTIFY/BURST=ONE/FLAG=ONE A.LIS;-1,A.LIS <-- ;-1 has the banners

*** Admin_info ***

_language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 06/06/85

Dates revised

**** BANNER3 ****

Write 10-line-high banners of up to 10 characters each (up to 3 banners per page).

Format:

RUN VSYS:BANNER3

banr1 banr2 banr3 /OUTPUT=file_spec <-- one or more of these,
end with empty line

Up to three banners can be printed on one page. This is designed for printing on wide paper.

See BANNER for 10-line-high banners entered one at a time for wide paper, and BANNER6 for 6-line-high banners for narrow paper.

*** Data ***

RUN VSYS:BANNER3

banr1 banr2 banr3 /OUTPUT=file_spec

You will be prompted for the message text. Each data entry has the form:

banr1 banr2 banr3 /OUTPUT=file_spec

where banri is up to 10 characters. The following are interpreted specially:

NODE - DT1 or DT2, as appropriate

DATE - the date in VAX/VMS format (dd-mmm-yyyy)

TIME - the time (hh:mm:ss)

Default: no defaults for banners; /OUTPUT=sys\$output

*** Qualifier ***

/OUTPUT

/OUTPUT=file_spec

Specifies the file into which the banners are to be written.

Default: /OUTPUT=SYSS\$OUTPUT for the first message.

The previous file_spec is used for succeeding messages (that is, messages are written to the same file until changed).

*** Examples ***

RUN VSYS:BANNER3

banr1 banr2 banr3 /OUTPUT=file_spec

- 1) Write the banners "MYOWN" and "LIBRARY", list the library and send to the printer.

```
RUN VSYS:BANNER3
```

```
MYOWN LIBRARY /OUTPUT=A.LIS
```

```
<cr>
```

```
<-- no more banners (empty line)
```

```
LIBRARY/LIST/FULL/OUTPUT=A.LIS myown_library <-- make library list
```

```
PRINT/NOTIFY/BURST=ONE/FLAG=ONE A.LIS;-1,A.LIS <-- ;-1 has the  
banners
```

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 06/26/85

Dates revised

***** BANNER6 *****

Write 6-line-high banners of up to 10 characters each (up to 6 banners per page).

Format:

RUN VSYS:BANNER6

```
banr1 banr2 banr3 banr4 banr5 banr6 /OUTPUT=file_spec <-- one or more  
                                         .  
                                         of these,  
                                         end with  
                                         empty line
```

Up to six banners can be printed on one page. This is designed for printing on narrow paper.

See BANNER for 10-line-high banners entered one banner at a time for wide paper, and BANNER3 for 10-line-high banners entered one page (3 banners) at a time for wide paper.

*** Data ***

RUN VSYS:BANNER6

banr1 banr2 banr3 banr4 banr5 banr6 /OUTPUT=file spec

You will be prompted for the message text. Each data entry has the form:

```
banr1 banr2 banr3 banr4 banr5 banr6  /OUTPUT=file_spec
```

where **banri** is up to 10 characters. The following are interpreted specially:

NODE - DT1 or DT2, as appropriate

DATE - the date in VAX/VMS format (dd-mmm-yyyy)

TIME - the time (hh:mm:ss)

Default: no defaults for banners; /output=sys\$output

*** Qualifier ***

/OUTPUT

/OUTPUT=file spec

Specifies the file into which the banners are to be written.

Default: /OUTPUT=SYSS\$OUTPUT for the first message.

The previous `file_spec` is used for succeeding messages (that is, messages are written to the same file until changed).

*** Examples ***

```
RUN VSYS:BANNER6
banr1 banr2 banr3 banr4 banr5 banr6 /OUTPUT=file_spec
```

- 1) Write the banners "MYOWN" and "LIBRARY", list the library and send to the printer.

```
RUN VSYS:BANNER6
MYOWN LIBRARY /OUTPUT=A.LIS
<cr>                                <-- no more banners (empty line)
LIBRARY/LIST/FULL/OUTPUT=A.LIS myown_library <-- make library list
PRINT/NOTIFY/BURST=ONE/FLAG=ONE A.LIS;-1,A.LIS <-- ;-1 has the
                                         banners
```

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 06/25/85

Dates revised

04/18/86 - change I/O from unit 5 to *

**** CALLTREE ****

This program produces a flow chart of subroutine and function calls for a FORTRAN program.

Format:

RUN VSYS:CALLTREE

CALLTREE operates off the listing files generated by FORTRAN compiler. Prior to running the program, you should compile all routines with (at minimum) the FORT/LIST/CROSS command.

CALLTREE will prompt for file names: first the output file, then the input file(s). You can enter the input file names in any order - the extension ".LIS" is automatically appended if not specified. At least one file MUST contain the main program identified in a PROGRAM statement.

The output from CALLTREE looks like this:

```
Call tree for program MAIN
  2....SUB1 <file1.lis>
  3.....SUBSUB2 <file2.lis>
  2....SUB2 <file1.lis>
  2....SUB3 <file3.lis>
  3.....SUBSUB4 <file3.lis>
  etc.,
```

where each level of indentation implies a lower level of subroutine call.

Following that there will be a section listing each file and the names of the routines contained in it. If the filename is "<System?>" then that is an unresolved reference (or possibly a system or library call).

Do NOT use CALLTREE if the program being analyzed contains recursive calls, as infinite looping can occur.

N.B. While CALLTREE was designed for version 3 of the compiler, it appears to work with version 4.

*** Sample_output ***

```
Call tree for program CALLTREE
  2....INIT <CALLTREE.LIS>
  3.....FOR$OPEN < >
  2....LISTEN <CALLTREE.LIS>
  3.....CH$RA1 (ARRAY,N) <LIB$CONSAD:SORTS.LIS>
  3.....ISALPHA (CHAR) <LIB$CONSAD:CHARSUBS.LIS>
  3.....ISDIGIT (CHAR) <LIB$CONSAD:CHARSUBS.LIS>
  3.....LIB$INDEX < >
```

```
3.....TRULEN (STRING) <LIB$CONSAD:CHARSUBS.LIS>
2.....PRINTEM <CALLTREE.LIS>
3.....TRULEN (STRING) <LIB$CONSAD:CHARSUBS.LIS>
2.....READEM <CALLTREE.LIS>
3.....FIND_NW (LINE,POS) <LIB$CONSAD:CHARSUBS.LIS>
3.....FIND_SP (LINE,POS) <LIB$CONSAD:CHARSUBS.LIS>
3.....FOR$OPEN < >
3.....ISALPHA (CHAR) <LIB$CONSAD:CHARSUBS.LIS>
3.....ISDIGIT (CHAR) <LIB$CONSAD:CHARSUBS.LIS>
3.....LIB$INDEX < >
3.....STR$UPCASE < >
3.....TRULEN (STRING) <LIB$CONSAD:CHARSUBS.LIS>
2.....SORTEM <CALLTREE.LIS>
3.....CHSRA2 (ARRAY,PTR,N) <LIB$CONSAD:SORTS.LIS>
```

LISTING OF ROUTINES IN EACH FILE

FILE<> CONTAINS ROUTINES:

```
FOR$OPEN (system ?)
LIB$INDEX (system ?)
STR$UPCASE (system ?)
```

FILE<CALLTREE.LIS> CONTAINS ROUTINES:

```
CALLTREE
INIT
LISTEM
PRINTEM
READEM
SORTEM
```

FILE<LIB\$CONSAD:CHARSUBS.LIS> CONTAINS ROUTINES:

```
FIND_NW (LINE,POS)
FIND_SP (LINE,POS)
ISALPHA (CHAR)
ISDIGIT (CHAR)
LOCATE (STR,POS,CHARS) (unreferenced)
RJUST (STRING) (unreferenced)
STREQ (S1,S2) (unreferenced)
TRULEN (STRING)
VERIFY (STR,POS,CHARS) (unreferenced)
```

FILE<LIB\$CONSAD:SORTS.LIS> CONTAINS ROUTINES:

```
CHSRA1 (ARRAY,N)
CHSRA2 (ARRAY,PTR,N)
I2SRA1 (ARRAY,N) (unreferenced)
I4SRA1 (ARRAY,N) (unreferenced)
I4SRA2 (ARRAY1,ARRAY2,N) (unreferenced)
I4SRA3 (ARRAY1,ARRAY2,ARRAY3,N) (unreferenced)
R4SRA1 (ARRAY,N) (unreferenced)
```

Language: Fortran 77

Author: Mike Shefler
Consad Research Corp.
121 N. Highland Ave.
Pittsburgh, PA 15206
(412) 363-5500

Date written:

Dates revised

10/01/85 - implemented on DTNSRDC VAXclutser without modification

**** CMP2FOR ****

Prepare compiler listing for printing on CDC (or any other printer) by changing <FF> in column 1 to '1', detabbing, and shifting lines 1 column to the right. The resulting file has the record attribute "Fortran carriage control" and can also be printed on a VAXcluster printer.

Format:

```
RUN VSYS:CMP2FOR      -or-      CMP2FOR == "$ VSYS:CMP2FOR"
infile outfile      CMP2FOR infile outfile
```

*** Parameters ***

infile is the specification of the input file which is a compilation listing.

Default file_type: .LIS

outfile is the resulting file after processing. It may be HFTed to the Mass Storage System and then MSFETChed on CDC and ROUTEd to a printer.

Default file_type: .LST

Default file_spec: infile.LST

*** Example ***

Compile MYPROG.FOR and print it on the CDC at remote site 142:

```
On VAXcluster:  FORTRAN /LIST /CROSS_REFERENCE /SHOW myprog
                  CMP2FOT:= $ VSYS:CMP2FOR      <-- only once per session
                  CMP2FOR myprog
                  (The output file will be in myprog.LST)
                  HFT ACCESS /PASSWORD=password
                  HFT STORE myprog.lst myprog
```

```
On CDC:          MSACCES,password
                  REQUEST,myprog,*Q
                  MSFETCH,myprog
                  ROUTE,myprog,DC=PR,FID=*xxxx,TID=142
                  MSPURGE,myprog
educue them to one statement each.
```

*** Admin_info ***

Language: DEC VAX/VMS Fortran 77

- 86/05/30

VAX

UTILITY

CMP2FOR

Page 2-12

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 02/13/86

Dates revised

**** COPYSF ****

Copy shifted file.

Format:

RUN VSYS:COPYSF

In response to the prompt

File?

enter

file_spec [/OUTPUT=] [/SHIFT=n_columns] [/CC | /NOCC]

*** Parameters ***

RUN VSYS:COPYSF file_spec [/OUTPUT=] [/SHIFT=n_columns] [/CC | /NOCC]

file_spec is the file to be copied shifted

*** Qualifiers ***

/OUTPUT

/OUTPUT=output-file-spec

If specified, the shifted file will be put into this file.

If omitted and the input file is SYSS\$INPUT, then the output will go to SYSS\$OUTPUT.

If omitted and the input file is not SYSS\$INPUT, then the output will go into the input filename with file type .LIS.

/SHIFT

/SHIFT=number_of_columns

Specify the number of columns to shift.

(Default: /SHIFT=1; maximum: /SHIFT=133)

/CC

/CC
/NOCC

Specify whether the input file has carriage control characters in column 1.

(Default: /NOCC)

*** Examples ***

- 1) Copy file myfile.for to myfile.lis shifting it 1 column to the right:

RUN VSYS:COPYSF
File? myfile.for

- 2) Copy file myfile.for to SYSSOUTPUT shifting 10 columns to the right:

RUN VSYS:COPYSF
File? MYFILE.FOR /OUTPUT=SYSSOUTPUT /SHIFT=10

- 3) Copy file MYFILE.DOC (with carriage control in column 1 of each record), centering the 72-column document on wide paper (output in myfile.lis):

RUN VSYS:COPYSF
File? MYFILE.DOC /SHIFT=25 /CC

*** Admin_info ***

Language: Fortran 77 extended

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 08/07/85

Dates revised

**** CPYEXT ****

Copy a file. Optionally, any number of fields within each line may be moved around. This is useful for removing the id fields from a source program, or re-arranging data fields, etc.

Format:

SET COMMAND VSYS:CPYEXT <-- only needed once during a batch job
or interactive session

CPYEXT infyl [outfyl] [/FIELDS=(from-to\at)]
[/KEEPCC]
[/LOG]

If infyl is omitted, you will be prompted for both infyl and outfyl.

*** Parameters ***

CPYEXT infyl [outfyl] [/FIELDS=(from-to\at)] [/KEEPCC] [/LOG]

infyl - The input file to be copied.

outfyl - The output file. If omitted, the next version of infyl
will be created.

*** Qualifiers ***

/FIELDS

/FIELDS=(from-to\at,...)

Specify one or more fields to be moved. If omitted, the first 72
columns are copied.

from - the starting column for extracting
(1-140; default: 1)

to - the ending column for extracting
(1-140; default: end of record; if all fields omitted,
default: 72)

at - the first column in the output record for the extracted field
(1-140; default: 1)

Notes: 1) To remove the ID field from a source program (columns 73-end),

omit this qualifier.

- 2) If TO < FROM, the field is reversed in the output record.

Default: /FIELD=(1-72\1)

/KEEPCC

/KEEPCC
/NOKEEPCC

If specified, column 1 (the carriage control character) is kept.

Default: /NOKEEPCC

/LOG

/LOG
/NOLOG

If specified, warning and informative messages will be issued.
If omitted, only fatal messages will be issued.

Default: /NOLOG

*** Examples ***

CPYEXT infyl [outfyl] [/FIELDS=(from-to\at)] [/KEEPCC] [/LOG]

- 1) Copy file MYPROG.FOR, removing the id field (columns 73 on), as the next version of the file.

SET COMMAND VSYS:CPYEXT <-- first time only

CPYEXT MYPROG.FOR

- 2) Move columns 10-72 of a document to columns 2 on and retain the carriage control characters (in column 1).

SET COMMAND VSYS:CPYEXT <-- first time only

CPYEXT old_file.doc new_file.doc /FIELDS=(10-72\2) /KEEPCC

- 3) Extract a Fortran source program from an output listing. Note that you will have to remove page heading lines yourself.

SET COMMAND VSYS:CPYEXT <-- first time only

CPYEXT myprog.LIS myprog.FOR /FIELDS=(9-80\1)

- 4) Swap the data in columns 1-10 and 21-30. Keep all other data.

SET COMMAND VSYS:CPYEXT <-- first time only

CPYEXT old_file_spec new_file_spec /F=(1,1-10\21,21-30\1)

The first field "1" copies the entire record as is.
The second field "1-10\21" moves columns 1-10 into 21-30.
The third field "21-30\1" moves columns 21-30 into 1-10.

- 5) Keep columns 21-24 in position, delete the rest of each record.

SET COMMAND VSYS:CPYEXT <-- first time only

CPYEXT old_file_spec new_file_spec /F=(21-24\21)

- 6) Reverse each 80-character record.

SET COMMAND VSYS:CPYEXT <-- first time only

CPYEXT old_file_spec new_file_spec /F=(80-1)

*** Admin_info ***

Language: VAX/VMS Fortran 77

Authors: Sharon E. Good - DTNSRDC Code 1892.1
David V. Sommer - DTNSRDC Code 1892.2

Date written: 06/05/85

Dates revised

**** HEAD ****

Display the first 22 lines of a file.

Format:

HEAD :== \$ VSYS:HEAD	-or-	RUN VSYS:HEAD
HEAD file-spec		file-spec

The heading will indicate if it is the whole file or just the first 22 lines.

*** Parameters ***

file-spec is the name of the file to be displayed.

Lines longer than 140 characters will be truncated without a message.

If lines exceed the page width, the display will roll off the top of the screen.

*** Admin_info ***

Language: Fortran 77 extended

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 03/21/86

Dates revised

**** LIST1 ****

This program is used to produce a centered list of a document or form (up to 90 characters per line and having carriage control in column 1). Optionally, the record-number-in-file, page number, line-number-on-page, and line lengths may be printed. Lines that are too long may also be flagged. Output may be for wide paper or, if the data lines are short enough, narrow paper.

Format:

RUN VSYS:LIST1

You will be prompted for:

Input file:

Enter the file-spec with or without qualifiers.

*** Qualifiers ***

The qualifiers may follow the command name or the file-spec.

/COUNTS

/COUNTS
/NOCOUNTS

Specifies whether the record number, page number, line on page, line length and too-long flag are to be printed.

Default: /COUNTS

/IGNORE

/IGNORE
/NOIGNORE

Determines whether lines with 'Q', 'R', 'S', 'T', or 'V' carriage control are to be ignored.

Default: /IGNORE

/LONG

/LONG=line_length

Specify the maximum line length. Lines longer than this are flagged with an asterisk '*'.

This has meaning only if /COUNTS is active.

Default: /LONG=72

/SHORT

/SHORT

/NOSHORT

When set, output is to be narrow instead of wide.

Default: /NOSHORT

*** Output_description ***

If /NOCOUNTS is active, the (wide) output is a centered list of the document or form using the carriage control in column 1. For narrow output, the list is shifted 17 columns to the right.

If /COUNTS is active, the output has 6 columns:

Wide output:

col	contents
---	-----
1	record number in the file
2	page number
3	line number on the page
4	the document or form
5	an asterisk is the line is too long (longer than LONG characters)
6	the length of the line

Narrow output:

col	contents
---	-----
1	record number in the file
2	page number
3	line number on the page
4	an asterisk is the line is too long (longer than LONG characters)
5	the length of the line
6	the document or form

If /IGNORE is active, all lines starting with 'Q', 'R', 'S', 'T', or 'V' are ignored, but are counted if /COUNTS is active.

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 03/05/85

Dates revised

07/22/85 - add /SHORT qualifier

**** LISTM ,****

List a file in multiple columns.

Format:

LISTM infyl [qualifiers]

*** Parameter ***

infyl - the file to be listed

*** Qualifiers ***

/COLUMNS

/COLUMNS=n - number of columns in the listing

(default: determined by /FROM, /TO, /WIDTH, /PW)

/DOUBLE

/DOUBLE - double spacing
/NODOUBLE - single spacing

(default: /NODOUBLE)

/FROM

/FROM=n - starting column of input file to list

(default: /FROM=1)

/HEADING

/HEADING - heading are to be printed
/NOHEADING - no headings are to be printed

(default: /HEADING)

If /HEADING is specified, the first line of each page has the date, time and file name, followed by a blank line. Then column markers for each column, followed by a blank line. The column markers also appear at the bottom of each page.

If /HEADING is not specified, only the selected data appears in the list.

/OUTPUT

/OUTPUT=file_spec - the file to hold the output list

(default: SYSS\$OUTPUT)

/PD

/PD=n - print density. one of:

8 or omitted - 8 lines per inch

other - 6 lines per inch

(default: /PD=6)

/PS

/PS=n - page size (number of lines to print per page (4 heading lines may also be printed)

(defaults: for /PD=8 /NODOUBLE /NOHEADER, /PS=82;
/PD=8 /NODOUBLE , /PS=78;
/PD=8 /DOUBLE /NOHEADER, /PS=41;
/PD=8 /DOUBLE , /PS=39;
/PD=6 /NODOUBLE /NOHEADER, /PS=60;
/PD=6 /NODOUBLE , /PS=56;
/PD=6 /DOUBLE /NOHEADER, /PS=30;
/PD=6 /DOUBLE , /PS=28)

/PW

/PW=n - page width

(max: 132)

(default: /PW=132)

/SEPARATION

/SEPARATION=n - number of blanks to separate columns in output list

(synonym: /BLANKS=)

(default: /SEPARATION=0)

/TO -

/TO=n - ending column of input to list

(default: /TO=80)

/WIDTH

/WIDTH=n - width of the input field

(default: /WIDTH=80)

(if only /TO or /WIDTH is specified, the other is calculated)

*** Output_description ***

Specified field of each record in the file is listed in columns. If headings are requested, each column has column markers (1...5...10..) at both the top and the bottom of each page.

*** Examples ***

- 1) List columns 1-20 of MYFILE.DAT in 6 columns with 2 blanks between columns:

LISTM MYFILE.DAT /WIDTH=20 /COLS=6 /SEP=2

- 2) List columns 73-80 in as many columns as will fit on a page 80 columns wide (uses defaults of /PS=56 /TO=80):

86/05/30

VAX

UTILITY

LISTM

Page 2-25

LISTM MYFILE.DAT /FROM=73 /PW=80 /SEP=5 .

*** Admin_info ***

Language: DEC VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 11/15/82 (CDC NOS/BE version)

Dates revised

01/22/86 - convert to VAXcluster

**** LISTN ****

Numbered list of a file. Optionally, the line lengths may be listed and lines longer than a specified length may be flagged.

Format:

RUN VSYS:LISTN

In response to the

Input (and output) files?

prompt, enter the name of the file to be listed, optional output file, and any qualifiers. The default output file is input_file.LIS.

An alternate way is to put "\$LISTN := \$VSY:LISTN" into your LOGIN.COM file and then type "LISTN infyl [outfyl] [qualifiers]" to use LISTN.

*** Parameters ***

LISTN infyl [outfyl]

infyl is the name of the file to be listed.

outfyl is the output file to contain the list. If omitted, the output will be written into file infyl.LIS.

*** Qualifiers ***

/LONG=n

If specified, lines longer than n character are flagged with an asterisk (*) before the line. Unflagged lines are preceded by a minus (-).

/LONG has effect only if /SHOW is active.

Default: /LONG=72

/SHOW

/SHOW
/NOSHOW

If specified, line lengths are listed and long lines are flagged.

Default: /NOSHOW

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 04/02/85

Dates revised

05/14/86 - add optional output file

**** LPL ****

List page lengths of a document or manual. For a manual, the page number, if in columns 65:75 of the top-of-page line, will also be listed.

Format:

RUN VSYS:LPL

In response to the

File?

prompt, enter the name of the file to be listed and any qualifiers.

*** Parameter ***

Program LPL has one parameter: the name of the file to be counted.

*** Qualifiers ***

/LINELEN

/LINELEN=n

Flag all lines longer than n characters. If n=0, long lines are not flagged.

Default: /LINELEN=72

/PAGELEN

/PAGELEN=n

Flag all pages with more than n lines.

Default: /PAGELEN=60

/SUP

/SUP
/NOSUP

If specified, lines with bad carriage control characters (not '1', ' ', '0', '+') in column 1 are not flagged.

Default: /NOSUP

/TYPE

/TYPE=type

Type of file to be processed.

/TYPE=DOCUMENT - file is a document

/TYPE=MANUAL - file is a manual (page number in columns 65:75
of the top-of-page line)

Default: /TYPE=DOCUMENT

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 03/27/85

Dates revised

02/19/86 - change width of overflow field to exact number of columns
required

**** LST2FOR ****

Copy a file changing the carriage control attribute from "carriage return" to "Fortran". This is needed so that files with carriage control characters in column 1 (such as a document prepared with an editor) will have the carriage control interpreted by the printer instead of just listed.

Format

RUN VSYS:LST2FOR

You will be prompted for

Input file? <-- If the file does not exist or if there is an error in the file specification, you will be asked to enter it again.

Output file?

A message summarizing the copy is typed at completion.

See procedure FOR2LST to go in the other direction.

*** Parameters ***

Input file - the name of the file having the "Carriage return carriage control" record attribute. Do "DIRECTORY/FULL file" to check.

Output file - the name of the file to contain the copy. It will have the "Fortran carriage control" attribute.

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: Sharon E. Good - DTNSRDC Code 1892.1

Date written: 03/15/85

Dates revised

**** MAKSUB ****

Generate a skeleton Fortran program or subprogram.

Format:

RUN VSYS:MAKSUB

In response to the prompt:

File?

enter

```
subfyl /CLEN=<character-function-length>
      /IMPL=<implicit-type>
      /NAME=<(sub)program-name>
      /TYPE=<(sub)program-type>
```

The generated (sub)program will be in file <subfyl>. If no type is specified, .FOR is used.

*** Parameter ***

RUN VSYS:MAKSUB subfyl

subfyl - file to contain the generated (sub)program
(If omitted, DUMMY.FOR is used.)

*** Qualifiers ***

/CLEN

/CLEN=n For /TYPE=C, the length of character function.
(default: /CLEN=(*))

/IMPL

/IMPL=i This specifies the kind of IMPLICIT statement, if any, is to appear in the subprogram. 'i' is one of:

C	- IMPLICIT CHARACTER (A-Z)
I	- IMPLICIT INTEGER (A-Z)
L	- IMPLICIT LOGICAL (A-Z)
N	- IMPLICIT NONE

R - IMPLICIT REAL (A-Z)
X - IMPLICIT COMPLEX (A-Z)
other - no IMPLICIT statement

(default: /IMPL=N)

/NAME

/NAME=name Name of the generated subprogram.

(default: /NAME=<subfyl-filename>)

/TYPE

/TYPE=t Type of (sub)program to be generated. 't' is one of:

C - character function
I - integer function
L - logical function
P - program
R - real function
S - subroutine
X - complex function
other - error

(default: /TYPE=S)

*** Examples ***

RUN VSYS:MAKSUB

Type "RUN VSYS:MAKSUB". Then in response to the 'File?' prompt:

- 1) To generate logical function MYLOG in file MYLOG.FOR with IMPLICIT NONE:

mylog/TYPE=L

- 2) To generate character * (*) function MYCHAR in file C.FOR with IMPLICIT NONE:

c/TYPE=c/NAME=mychar

- 3) To generate character * 25 function MYCH25 in file MYCH25.SUB with no IMPLICIT statement:

mych25.sub /TYPE=c /CLEN=25 /IMPL=omit

- 4) To generate subroutine MYSUB in file MYSUB.FOR with
IMPLICIT INTEGER (A-Z):

mysub /IMPL=i

- 5) To generate program MYPROG in file MYPROG.FOR with IMPLICIT NONE:

myprog /T=p

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 03/85

Dates revised

05/15/85 - fix problem with /CLEN

**** TAIL ****

Display the last 22 lines of a file.

Format:

TAIL :== \$ VSYS:TAIL	-or-	RUN VSYS:TAIL
TAIL file-spec		file-spec

The heading will indicate if it is the whole file or just the last 22 lines.

*** Parameters ***

file-spec is the name of the file to be displayed.

Lines longer than 140 characters will be truncated without a message.

If lines exceed the page width, the display will roll off the top of the screen.

*** Admin_info ***

Language: Fortran 77 extended

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 12/12/85

Dates revised

**** UNBLANK ****

Removes trailing blanks from a file. This is useful for shortening files moved to VMS from another computer.

Format:

RUN VSYS:UNBLANK

You will be prompted for:

Input,output files:

If the input-file is missing, both the input and output files are requested separately.

Input file: <-- enter the input file name

Output file: <-- enter the output file name -or-
blank (to use the input file name)

If the output-file is missing, the input-file is used, thus creating a new version of the file.

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 02/07/85

Dates revised

**** UNDECK ****

Break up a CDC UPDATE source file into separate files.

Format:

RUN VSYS:UNDECK

You will be prompted for:

Input file: <-- no default
Output subdirectory: <-- default: current directory
Keep all created files? <-- no default
*DECK filetype: <-- default: .FOR
*COMDECK filetype: <-- default: .INC

For help on any prompt, enter a question mark (?).

The name of each new file is displayed. If you answered NO to "Keep all created files?", you will be asked as each file is created if you want to keep it.

Each *DECK encountered will create a new file:

[.<subdirectory>]<deck_name>.<deck_filetype>

Each *COMDECK encountered will create a new file:

[.<subdirectory>]<comdeck_name>.<comdeck_filetype>

Each *CALL encountered will be converted to:

INCLUDE '<comdeck_name>.<comdeck_filetype>'

*** Input_file ***

The first prompt is for the name of the file containing the CDC UPDATE source data. You will receive an error message if the file does not exist or if there is a file specification error. You must then re-enter the file name.

The full file specification is echoed.

For some help about this parameter, enter a "?".

*** Output_subdirectory ***

The second prompt is for the subdirectory to hold the created files. It

is specified as "[.subdirectory_name]", You will receive an error message if there is a syntax error. You must then re-enter the subdirectory.

To put the created files into the current directory, enter a carriage return or "[]".

The output subdirectory is echoed.

For some help about this parameter, enter a "?".

*** Keep_all_files ***

The third prompt asks if you want to keep all the files which will be created.

To keep all of them, enter YES, KEEP, ALL, or TRUE. Only the first letter is tested and may be upper or lower case.

To have the program ask you if you want to keep each file as it is created, enter NO or FALSE. Only the first letter is tested and may be upper or lower case.

A message in English echoes your response.

For some help about this parameter, enter a "?".

If you answered NO (or FALSE), then as each file is created, you will be asked "Keep?". Respond with one of the following:

YES or KEEP	or TRUE	- to keep the file
NO	or DELETE or FALSE	- to delete (ignore) the file
ALL or CONTINUE		- to keep the file and all following files
QUIT		- to delete (ignore) the file and end the program

Again, only the first letter is tested and may be in upper or lower case and a "?" will give you some help.

*** DECK_filetype ***

The fourth prompt is for the file type to be used in the specification of the file from a *DECK. This is up to 39 alphanumeric character and may be preceded by a period (.). If there is an error in your entry, you will receive a message and must re-enter the *DECK filetype.

The default (if you enter just a carriage return) is ".FOR".

The *DECK file type is echoed.

For some help about this parameter, enter a "?".

*** COMDECK_filetype ***

The fifth prompt is for the file type to be used in the specification of the file from a *COMDECK. This is up to 39 alphanumeric character and may be preceded by a period (.). If there is an error in your entry, you will receive a message and must re-enter the *COMDECK filetype.

The default (if you enter just a carriage return) is ".INC" to indicate an INCLUDE module.

The *COMDECK file type is echoed.

For some help about this parameter, enter a "?".

*** Admin_info ***

Language: VAX/VMS Fortran 77

Author: David V. Sommer - DTNSRDC Code 1892.2

Date written: 02/12/85

Dates revised

- 10/25/85 - rewrite to include more analysis of user responses
- add conversion of *CALL to an INCLUDE
- change default *COMDECK file type from .FCM to .INC

**** VLASER ****

Program to combine one or more files into a single file for processing on the Xerox 8700. Xerox DJDE records are generated, if they are not already in the file. (A DJDE record contains Xerox 8700 control information. Such records are not printed, but dynamically modify the printing environment for the records which are subsequently printed.)

Format:

RUN VSYS:VLASER

You will be prompted for all required information.

The output is a file XRXTAPE.DAT which may be put onto tape using procedure @VSYS:CALCD2T.

[[lmIf you wish to go directly to tape, use @VSYS:WXTAPE, which will mount the [Om
[[lmtape and run this program. Type HELP @PROCFIL WXTAPE. [Om

Files with the record attribute "Fortran carriage control" are assumed to have carriage control characters in column 1. Files with the record attribute "carriage return carriage control" are assumed to have data starting in column 1. If your file does not have the correct record attribute, use program VSYS:LST2FOR (HELP LST2FOR) or procedure @VSYS:FOR2LST (HELP FOR2LST) or procedure @VSYS:NONE2LST (HELP NONE2LST) to change it.

*** Data_entries ***

If you need help for any question asked by VLASER, enter ? or HELP.

The response to every prompt (except file name) may be entered in upper or lower case and may be abbreviated by as few as one character.

Paper: - Enter 3-HOLE or PLAIN.
This will apply to the entire run.
(Default: 3-HOLE)

Duplex for job? - Enter YES if you want DUPLEX=YES for all files.
Enter NO if you want DUPLEX=NO for all files.
Enter BOTH if you want to be asked for each file.
If you have only one file to process, it doesn't matter how you respond.
(Default: YES)

File: - Enter the name of the (next) file to be copied.
Enter CTRL-Z or <cr> to indicate that no more files are to be processed.

Is/are the DJDE record(s) already in the file?

- Respond YES or NO.

<for all but the first file>

The last DJDE was:

<djde record>

Do you want to use the same one?

- Respond YES or NO. If NO, you will be prompted for each DJDE parameter.

<prompts for DJDE parameters>

Xerox jobname: - 6 characters maximum

(Computer Center-defined jobnames are recognized. If you type some other jobname, you will be asked to verify it. This will give you an opportunity to correct a misspelling.)
(Default: STDLND)

Number of copies:

- up to 3 digits
(default: 001)

Duplex?

- Asked only if you responded BOTH to the second prompt.
- Enter YES or NO
(Default: YES)

Forms:

- Enter the name of the dynamic forms overlay to be used for this file.
(Computer Center-defined forms names are recognized. If you type some other jobname, you will be asked to verify it. This will give you an opportunity to correct a misspelling.)
(Default: no FORMS= parameter is generated)

Do you want SIDE=NUFRONT?

- Respond YES if you want to force the next page to be printed on a new sheet of paper.
Respond NO if you want to let the Xerox 8700 decide for itself.
(Default: YES)

Is this DJDE to be used for all remaining files?

- Enter YES or NO. If you do not answer YES, you will be asked for each file.
(If you have only file to process, it doesn't matter how you respond.)

<for all but the first file>

Do you want a banner page for this file?

- Enter YES or NO. The first banner page will be blue. Other banner pages will be white, but will have a border on both sides to help locate it.

** Paper **

Your Xerox 8700 output may be on plain paper or 3-hole punched paper.

Enter either 3-HOLE (or <cr>) or PLAIN.

** Duplex **

DUPLEX=YES means print on both sides of each sheet of paper.

DUPLEX=NO means print on only one side (simplex).

We do NOT recommend mixing duplex and simplex printing within a given Xerox job because the Xerox 8700 operator must change the paper orientation for each duplex/simplex change and the first page of the new setting may not be what you want.

You are asked at the start if you want everything to be done duplex or simplex. If you really want to mix them, enter BOTH to the first duplex prompt. You will then be asked each time we create a new DJDE record.

** File_name **

The name of the (next) file to be processed is any valid file specification.

** In-file_DJDE **

If the DJDE records are already in the file, respond YES when asked.

If they are not in the file, respond NO and we will create one.

** Use_last_DJDE **

If you have more than one file and the DJDE record(s) are not in the next file, you are given the opportunity to use the last DJDE record we generated. Note that I only know the last DJDE record we created, not any which may be embedded in an earlier file.

** Job_name **

The Xerox jobname is the name of the job on the Xerox 8700 which will be

- used to process the file.

The job names I recognize are listed below. If you type something else (whether a different valid job name or a misspelling) I will ask you about it. If it is valid, enter it again as verification. If it was a misspelling, enter the correct spelling. Entering just <cr> will indicate the default, which is STDLND.

Landscape refers to a page orientation where the printed lines are parallel to the long side of the page.

Portrait refers to a page orientation where the printed lines are parallel to the short side of the page.

2-up is a term used to denote 2 pages of standard printout appearing on one side of a page in a portrait orientation.

2-across is a term used to denote two portrait pages appearing on one side of a page in a landscape orientation.

Landscape:

- STDLND - standard landscape job (default),
 - carriage control (CC) character plus 136 characters by 64 lines
 - (good for standard printouts)
- LN8LND - 8 lpi landscape job
 - CC character + 136 data characters by 86 lines
- 2ACLND - 2-across landscape job
 - 2 * (CC character + 66 data characters) by 60 lines
- BANNER - shaded border both sides
 - two pages must be provided, each starting with a '1' in column 1
 - a '2' in column 138 will print large letters

Portrait:

- STDPRT - standard portrait job

- CC character + 72 data characters by 64 lines
- ST3PRT - 3rd standard portrait job
 - CC character + 94 data characters by 64 lines
(very large margins)
- ST4PRT - 4th standard portrait job
 - CC character + 87 data characters by 64 lines
(general documentation; manuals)
- DOCPRT - document/manual portrait job
 - CC character + 87 data characters by 64 lines
(A '.' in column 76 of top-of-page line forces page onto front side)
- VAXDOC - like ST4PRT, except each page has exactly 66 lines
(blank carriage control on all lines)
- DD1473 - Standard Form DD-1473
 - both sides of the form are required (each begins with a '1' in column 1)

Portrait (2-up):

- 2UPPRT - 2-up portrait job
 - CC character + 136 data characters by 128 lines
(good for compact storage of standard printout)
- 2U8PRT - (8 lpi) - 2-up portrait job
 - CC character + 136 data characters by 176 lines
(good for compact storage of standard printout)
- 2UCPRT - 2-up portrait job
 - CC character + 136 data characters by 132 lines
(like 2UPPRT, but continuous output;
good for 2-page printer plots)

* BANNER_sample *

The following is the standard first banner for each job:

```
1...5...10...15...20...25...30...35...40...45...50...55...60...65...70
<DJDE> JDE=BANNER,OTEXT='<<<  LOAD 3-HOLE PAPER - DUPLEX  >>>,WAIT...
```

```
1
```

```
-
```

```
-
```

```
-
```

```
DAVID W. TAYLOR NAV ...
      BETHESDA, MAR ...
```

```
-
```

```
-
```

```
-
```

```
-
```

```
-
```

AMDS

29-AUG-1985

XEROX 8700

...

<<col 138>> 2

1

**** Copies ****

The number of copies is 1-3 digits. Leading spaces and zeros are ignored. If you want only one copy, enter just <cr>.

**** Overlay_forms ****

If a previous DJDE specified a FORMS parameter and you wish to use no special forms, enter NONE.

If you want the current form to be used, enter just <cr> and the FORMS parameter will not be included in the generated DJDE.

The forms names I recognize are listed below. If you type something else (whether a different valid forms name or a misspelling), I will ask you about it. If it is valid, enter it again as verification. If it was a misspelling, enter the correct spelling.

Landscape refers to a page orientation where the printed lines are parallel to the long side of the page.

Portrait refers to a page orientation where the printed lines are parallel to the short side of the page.

2-up is a term used to denote 2 pages of standard printout appearing on one side of a page in a portrait orientation.

NONE no forms overlay

Landscape:

FRAME1 box around page (1 solid line)
FRAME2 box around page (2 solid lines)
FRAME3 box around page (3 solid lines)
FRAME4 box around page (1 solid/shaded line)

2UPL1 solid line dividing two pages
2UPB1 solid boxes drawn around two pages
2UPL2 double solid lines dividing two pages

Portrait:

FRAMP1 box around page (1 solid line)
FRAMP2 box around page (2 solid lines)
FRAMP3 box around page (3 solid lines)
FRAMP4 box around page (1 solid/shaded line)

2UP3L triple solid lines dividing two pages
2UPSL single shaded line dividing two pages
2UPSB shaded bar dividing two pages
2UPB2 two separate solid line on a page

WEEKLY checklist by days

**** New-page_force ****

If you want to force the first page of the file onto a new sheet of paper, respond YES when asked "Do you want SIDE=NUFRONT?"

If you want to let the Xerox determine where it should be printed, respond NO.

Note that if you mix duplex and simplex, each time you change, the next page will be forced onto a new sheet.

**** All_files_DJDE ****

After we have created a new DJDE, you will be asked if you want it to be used for all remaining files. Respond YES or NO as you wish.

You will also be asked this if you are re-using the last DJDE.

Of course, if you have only one file, it doesn't matter which response you give.

*** Admin_info ***

Language: VAX/VMS Fortran 77

Authors: David V. Sommer - DTNSRDC Code 1892.2
Sharon E. Good - DTNSRDC Code 1892.2

Date written: 08/29/85

Dates revised

***** Index *****

BANNER	2-2
Data	2-2
Qualifiers	2-2
/NEWPAGE	2-2
/OUTPUT	2-2
Examples	2-3
Admin_info	2-3
 BANNER3	2-4
Data	2-4
Qualifier	2-4
/OUTPUT	2-4
Examples	2-4
Admin_info	2-5
 BANNER6	2-6
Data	2-6
Qualifier	2-6
/OUTPUT	2-6
Examples	2-7
Admin_info	2-7
 CALLTREE	2-8
Sample_output	2-8
Admin_info	2-9
 CMP2FOR	2-11
Parameters	2-11
Example	2-11
Admin_info	2-11
 COPYSF	2-13
Parameters	2-13
Qualifiers	2-13
/OUTPUT	2-13
/SHIFT	2-13
/CC	2-14
Examples	2-14
Admin_info	2-14
 CPYEXT	2-15
Parameters	2-15
Qualifiers	2-15
/FIELDS	2-15
/KEEPCC	2-16
/LOG	2-16
Examples	2-16
Admin_info	2-17
 HEAD	2-18

Parameters	2-18
Admin_info	2-18
LIST1	2-19
Qualifiers	2-19
/COUNTS	2-19
/IGNORE	2-19
/LONG	2-19
/SHORT	2-20
Output_description	2-20
Admin_info	2-20
LISTM	2-22
Parameter	2-22
Qualifiers	2-22
/COLUMNS	2-22
/DOUBLE	2-22
/FROM	2-22
/HEADING	2-22
/OUTPUT	2-23
/PD	2-23
/PS	2-23
/PW	2-23
/SEPARATION	2-24
/TO	2-24
/WIDTH	2-24
Output_description	2-24
Examples	2-24
Admin_info	2-25
LISTN	2-26
Parameters	2-26
Qualifiers	2-26
/LONG=n	2-26
/SHOW	2-26
Admin_info	2-27
LPL	2-28
Parameter	2-28
Qualifiers	2-28
/LINELEN	2-28
/PAGELEN	2-28
/SUP	2-28
/TYPE	2-29
Admin_info	2-29
LST2FOR	2-30
Parameters	2-30
Admin_info	2-30
MAKSUB	2-31
Parameter	2-31
Qualifiers	2-31
/CLEN	2-31
/IMPL	2-31

/NAME	2-32
/TYPE	2-32
Examples	2-32
Admin_info	2-33
TAIL	2-34
Parameters	2-34
Admin_info	2-34
UNBLANK	2-35
Admin_info	2-35
UNDECK	2-36
Input_file	2-36
Output_subdirectory	2-36
Keep_all_files	2-37
DECK_filetype	2-37
COMDECK_filetype	2-38
Admin_info	2-38
VLASER	2-39
Data_entries	2-39
Paper	2-41
Duplex	2-41
File_name	2-41
In-file_DJDE	2-41
Use_last_DJDE	2-41
Job_name	2-41
BANNER_sample	2-43
Copies	2-44
Overlay_forms	2-44
New-page_force	2-45
All_files_DJDE	2-45
Admin_info	2-46

Initial Distribution

Copies:

12 Director
Defense Technical Information Center (DTIC)
Cameron Station
Alexandria, Virginia 23314

Center Distribution

Copies:

1	18/1809	Gleissner, G. H.
1	1805	Cuthill, E. H.
2	1809S	
1	182	Camara, A. W.
1	184	Schot, J. W.
1	185	Schaffran, R.
1	187	Zubkoff, M. J.
1	189	Gray, G. R.
1	189.2	
1	189.3	Morris, J.
20	1892.1	Strickland, J. D.
10	1892.2	Sommer, D. V.
1	1892.3	Minor, L. R.
1	1894	
1	1896	Glover, A.
1	1896.2	Dennis, L.
1	522	TIC (C)
1	522.2	TIC (A)
1	93	Patent Counsel

DTNSRDC ISSUES THREE TYPES OF REPORTS

1. DTNSRDC REPORTS, A FORMAL SERIES, CONTAIN INFORMATION OF PERMANENT TECHNICAL VALUE. THEY CARRY A CONSECUTIVE NUMERICAL IDENTIFICATION REGARDLESS OF THEIR CLASSIFICATION OR THE ORIGINATING DEPARTMENT.

2. DEPARTMENTAL REPORTS, A SEMIFORMAL SERIES, CONTAIN INFORMATION OF A PRELIMINARY, TEMPORARY, OR PROPRIETARY NATURE OR OF LIMITED INTEREST OR SIGNIFICANCE. THEY CARRY A DEPARTMENTAL ALPHANUMERICAL IDENTIFICATION.

3. TECHNICAL MEMORANDA, AN INFORMAL SERIES, CONTAIN TECHNICAL DOCUMENTATION OF LIMITED USE AND INTEREST. THEY ARE PRIMARILY WORKING PAPERS INTENDED FOR INTERNAL USE. THEY CARRY AN IDENTIFYING NUMBER WHICH INDICATES THEIR TYPE AND THE NUMERICAL CODE OF THE ORIGINATING DEPARTMENT. ANY DISTRIBUTION OUTSIDE DTNSRDC MUST BE APPROVED BY THE HEAD OF THE ORIGINATING DEPARTMENT ON A CASE-BY-CASE BASIS.

END

12-86

DTIC